

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 10-078932
 (43)Date of publication of application : 24.03.1998

1)Int.CI.
 G06F 13/00
 G06F 9/06
 G06F 9/445
 G06F 13/14
 G06F 15/00

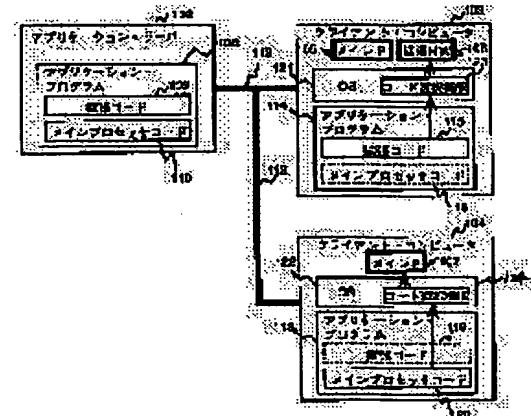
1)Application number : 08-233913
 (71)Applicant : HITACHI LTD
 2)Date of filing : 04.09.1996
 (72)Inventor : SUZUKI TAKASHI

1) RECONFIGURABLE NETWORK COMPUTER

2)Abstract:

PROBLEM TO BE SOLVED: To enable a user to receive the service of higher quality that requires the computational complexity without purchasing a new computer of higher performance nor adding any hardware to extend the computer function.

SOLUTION: A client 103 has the extension hardware 106 which can occasionally change and reconfigure the computer function via a program. An application program 108 stored in a server 102 includes an extension code 109 to reconfigure the hardware 106 of the client 103 in addition to a main processor code 110. Furthermore, the server 102 or the client 103 has a code selection function 123 to decide the presence or absence and the type of the hardware 106 and also to take a proper code out of both codes 109 and 110.



GAL STATUS

[date of request for examination] 06.05.2003
 [date of sending the examiner's decision of rejection]
 [end of final disposal of application other than the examiner's decision of rejection or application converted registration]
 [date of final disposal for application]
 [patent number]
 [date of registration]
 [number of appeal against examiner's decision of rejection]
 [date of requesting appeal against examiner's decision of rejection]

(19) 日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平10-78932

(43) 公開日 平成10年(1998)3月24日

(51) Int.Cl. ^a	識別記号	序内整理番号	F I	技術表示箇所
G 06 F 13/00	3 5 7		G 06 F 13/00	3 5 7 Z
9/06	4 1 0		9/06	4 1 0 S
9/445			13/14	3 3 0 A
13/14	3 3 0		15/00	3 1 0 J
15/00	3 1 0		9/06	4 2 0 J

審査請求 未請求 請求項の数14 O L (全 17 頁)

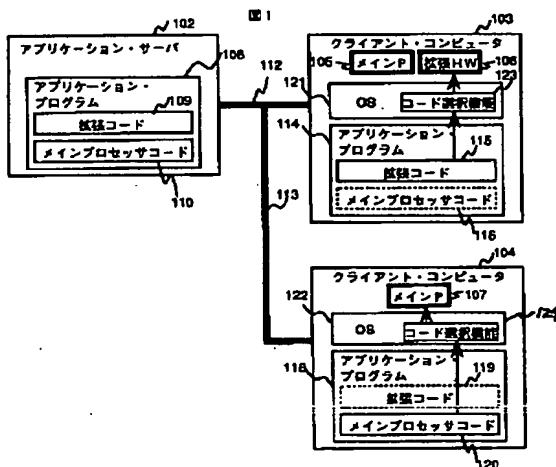
(21) 出願番号	特願平8-233913	(71) 出願人	000005108 株式会社日立製作所 東京都千代田区神田駿河台四丁目6番地
(22) 出願日	平成8年(1996)9月4日	(72) 発明者	鈴木 敬 東京都国分寺市東恋ヶ窪一丁目280番地 株式会社日立製作所中央研究所内
		(74) 代理人	弁理士 小川 勝男

(54) 【発明の名称】 リコンフィグラブル・ネットワークコンピュータ

(57) 【要約】

【課題】利用者が、より高性能なコンピュータに買い換えるか、機能拡張を行うためのハードウェアを追加するかを行うことなしに、計算量を必要とする、より高度なサービスを受けられるようにする。

【解決手段】クライアントは、プログラムにより機能を随时変更し再構成することが可能な拡張ハードウェア106を搭載し、サーバに格納されたアプリケーションプログラム108は、メインプロセッサコード110の他に前記クライアントの拡張ハードウェアを再構成するための拡張コード109を含み、さらに、クライアントの拡張ハードウェアの有無または種類を判断し、前記拡張コードとメインプロセッサコードの中から適切なコードを取り出すコード選択機能123を、サーバまたはクライアントのいずれかに有する。



1

【特許請求の範囲】

【請求項1】ネットワークにより接続された複数の種類の異なるコンピュータにより構成され、そのうちの少なくとも1つのコンピュータがアプリケーションプログラムを配布するサーバであり、残りのコンピュータが前記アプリケーションプログラムをダウンロードし、実行するクライアントとなるコンピュータであるシステムにおいて、複数の前記クライアントの一部は、プログラムにより機能を随時変更し再構成することが可能な拡張ハードウェアを搭載し、サーバに格納された、アプリケーションプログラムは、前記クライアントの拡張ハードウェアを再構成するためのプログラムコードである拡張コードと、前記クライアントのメインのプロセッサ上で実行されるべきコードであるメインプロセッサコード、の2種のコードを含み、さらに、各クライアントに前記拡張ハードウェアが実装されているか否か、また拡張ハードウェアの種類を判断し、前記アプリケーションプログラムの拡張コードと、メインプロセッサコードの中から適切なコードを取り出すコード選択機能を、ホストまたは前記拡張ハードウェアを搭載したクライアントコンピュータのいずれかに有し、拡張ハードウェアを搭載したクライアント上では、クライアントのハードウェアをアプリケーションプログラムに適した構成に随時再構成し実行し、また拡張ハードウェアを搭載しないクライアント上ではメインプロセッサコードで実行することを特徴とするリコンフィグラブル・ネットワークコンピュータ。

【請求項2】請求項1において、拡張ハードウェアで実現する機能を、後から動的に追加／削除が可能なクライアント上のOSの拡張機能として実現し、前記OSの拡張機能が、請求項1におけるアプリケーションプログラムと同様に拡張ハードウェア用の拡張コードとメインプロセッサコードの2種を含み、アプリケーションプログラムが起動時にOSに対し、処理中に利用する前記OSの拡張機能の種類を登録し、前記OSは既にその拡張機能が、前記クライアント上に存在する場合にはそれを用い、存在しない場合はネットワーク上のサーバから必要とする拡張機能を転送し利用するOS追加機能を、ホストまたは拡張ハードウェアを搭載したクライアントコンピュータのいずれかに有することを特徴とするリコンフィグラブル・ネットワークコンピュータ。

【請求項3】請求項1において、拡張ハードウェアで実現する機能を、実行時にアプリケーションプログラムに動的に追加されるライブラリとして実現し、前記動的ライブラリが、請求項1におけるアプリケーションプログラムと同様に拡張ハードウェア用の拡張コードとメインプロセッサコードの2種を含み、アプリケーションプログラムが起動時にOSに対し、処理中に利用する前記動的ライブラリの種類を登録し、前記OSは既にその動的ライブラリが、前記クライアント上に存在する場合にはそれを用い、存在しない場合はネットワーク上のサーバ

10

2

から必要とする動的ライブラリを転送し利用する動的ライブラリ追加機能を、ホストまたは拡張ハードウェアを搭載したクライアントコンピュータのいずれかに有することを特徴とするリコンフィグラブル・ネットワークコンピュータ。

20

【請求項4】請求項1、請求項2または請求項3において、メインプロセッサ用のコード、拡張ハードウェア用のコードが一体となっているのではなく、アプリケーションプログラムまたはOSの拡張機能または動的ライブラリ毎に、個々のコードをホストコンピュータ上に備えたことを特徴とするリコンフィグラブル・ネットワークコンピュータ。

30

【請求項5】請求項1、請求項2、請求項3または請求項4において拡張ハードウェアをプログラムを実行するプロセッサにより構成し、拡張コードを前記プロセッサの命令コードとすることを特徴とするリコンフィグラブル・ネットワークコンピュータ。

40

【請求項6】請求項1、請求項2、請求項3または請求項4において拡張ハードウェアをプログラマブルロジックを用いて構成し、拡張コードを前記プログラマブルロジックを再構成するためのコンフィギュレーション・データとすることを特徴とするリコンフィグラブル・ネットワークコンピュータ。

50

【請求項7】請求項6において、複数個のクライアントの拡張ハードウェアがプログラマブルロジックであり、その構造がそれぞれ異なる場合に、拡張コードは、適当なゲート数と入出力端子数の論理回路の機能をブール式等で記述した基本モジュールとそれらの接続関係を表現したコードであり、前記基本モジュールをそれぞれプログラマブルロジックの基本プログラム単位に割り付ける機能と、複数のプログラマブルロジックチップにまたがる大きな拡張コードの場合は、基本モジュールを接続の度合に応じて分割し、各プログラマブルロジックチップに配置配線し端子の機能を決定する機能と、プログラマブルロジック内の基本プログラム単位への基本モジュールの配置配線をする機能をサーバまたはクライアント上に持つことを特徴とするリコンフィグラブル・ネットワークコンピュータ。

60

【請求項8】請求項5または請求項6において、複数個のクライアントの拡張ハードウェアがそれぞれ異なる構造を持つ場合に、拡張コードは前記複数の構造の個々のコードをすべて含み、ハードウェア構造に合わせたコードを選択する機能を前記コード選択機能が持つことを特徴とするリコンフィグラブル・ネットワークコンピュータ。

70

【請求項9】請求項1、請求項2、請求項3、請求項4、請求項6、請求項7または請求項8において、メインプロセッサがプログラマブルロジックで構成されており、プログラマブルロジックの利用されていない論理資源を拡張ハードウェアとして利用することを特徴とする

リコンフィグラブル・ネットワークコンピュータ。

【請求項10】請求項1、請求項2または請求項3において、クライアントのOSが、拡張ハードウェアを利用する複数のアプリケーションを同時に実行できるように、各アプリケーションが利用する拡張ハードウェアのハードウェア資源を管理し、必要ななくなったハードウェア資源を別のアプリケーションプログラムのために再利用し、各アプリケーションをメインプロセッサで実行するか、拡張ハードウェアで実行するかを決定する拡張ハードウェア管理機能と前記拡張ハードウェアに入り切らないアプリケーションプログラムの拡張コードを時分割で入れ替えるコード入れ替え機能を持つことを特徴とするリコンフィグラブル・ネットワークコンピュータ。

【請求項11】請求項10において、クライアント上で実行されるアプリケーションプログラム毎に適宜設定される実行プライオリティ値、前記アプリケーションプログラムをすべてメインプロセッサで処理するために必要とする正規化されたメインプロセッサの処理能力値、前記アプリケーションプログラムを拡張ハードウェアで処理する場合に必要な正規化されたメインプロセッサと拡張ハードウェアの処理能力値、前記アプリケーションプログラム毎に必要とする拡張ハードウェアの正規化されたハードウェア資源使用量、前記クライアントのメインプロセッサの正規化された処理能力値、前記クライアントの拡張ハードウェアの正規化された処理能力値、前記クライアントの拡張ハードウェアの正規化されたハードウェア資源量、および前記クライアントの拡張ハードウェアのコードを入れ替えるために必要な正規化された処理能力値を元に、前記クライアントの拡張ハードウェアの持つハードウェア資源に入り切らない複数のアプリケーションプログラムに対して、優先的に拡張ハードウェアを利用するアプリケーションプログラム、時分割で拡張コードを入れ替えて使用するアプリケーションプログラム、メインプロセッサ用コードで処理するアプリケーションプログラムを選択する拡張ハードウェア管理機能を持つことを特徴とするリコンフィグラブル・ネットワークコンピュータ。

【請求項12】請求項10において、複数のアプリケーションが同時に同じ拡張コードを拡張ハードウェアで利用する場合に、前記拡張ハードウェアが、複数の内部状態を同時に保持し、かつそれらを切り替える機能を持ち、内部状態のみを時分割で切り替えて機能を共有することを特徴とするリコンフィグラブル・ネットワークコンピュータ。

【請求項13】請求項10において、拡張ハードウェアが、複数の内部状態と複数の構成データを同時に保持し、かつそれらを切り替える機能を持ち、内部状態と構成データを時分割で切り替えて使用できることを特徴とするリコンフィグラブル・ネットワークコンピュータ。

【請求項14】請求項13において、拡張ハードウェア

が、複数の内部状態と複数の構成データを同時に保持し、かつそれらが外部から与えられる識別番号と、あらかじめ構成データ、内部状態毎に割り当てられた識別番号と一致する部分のみが選択される機能を持つことを特徴とするリコンフィグラブル・ネットワークコンピュータ。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、ハードウェアを再構成可能とする機能を持つネットワークに接続した計算機システムに関する。

【0002】

【従来の技術】例えば、(1)日経エレクトロニクス1996.2.12, No.655, pp.13-15に解説されている「ネットワークコンピュータ」は、OSも含めて各コンピュータ上で必要なアプリケーションは全てネットワーク上のサーバからダウンロードすることにより、ハードディスクなどの2次記憶装置を持たない、単純で安価なコンピュータを実現する。さらに、各ユーザが新しいアプリケーションソフトウェアが記録されているフロッピーディスクやCD-ROM等を市の店舗で購入し、各自のコンピュータにインストールするという繁雑な作業をしなくとも、常に最新のアプリケーションソフトウェアを利用できるようになる。

(2)また、日経コンピュータ1996.4.1,p.101-107に解説されている「Java言語」はソフトウェアを実行するCPUのアーキテクチャに依存しないインタプリタ型の言語であるため、コンピュータを選ばずにネットワーク上に接続する多種のコンピュータのどこでも実行可能であるという特徴を持つ。この特徴は(1)の「ネットワークコンピュータ」の考え方において、バイナリコードの互換性がないコンピュータが、ネットワークで接続されている環境においても、「ネットワークコンピュータ」の環境を実現できるという効果を生む。また、アプリケーションソフトウェアを開発するソフトウェア企業は、コンピュータの機種毎に異なるバイナリ形式のアプリケーションソフトウェアを用意する必要がなくなる。

(3)更に、日経コミュニケーション1996.1.15, pp.79-83や、日経エレクトロニクス1996.1.15, p.67-95に解説されているように、世界的なネットワークであるインターネット上では音声や動画像をリアルタイムで配信する技術が試されている。この技術により、コンピュータネットワーク上でラジオやテレビといった放送サービスを実現できる。従来、インターネット上では、テキスト、静止画像、あらかじめ記録された音声や動画像の再生、等の多種の形態の情報を通信することが可能であった。これらの情報を組み合わせた複合情報(ハイパー・テキスト)を閲覧するサービスであるワールド・ワイド・ウェブ(WWW)では、ユーザはハイパー・テキス

トを閲覧するアプリケーション（ブラウザ）を利用して、サービスを利用する。

【0005】リアルタイムの音声／動画像の配信は、従来のブラウザの拡張機能（プラグイン）として、サービスを行う企業あるいは技術を開発した企業から、ネットワークを通して、無料、あるいは安価で提供されている。このサービスを受けたいユーザはネットワークから必要なプラグインをダウンロードし、自分のコンピュータにインストールすることにより、直ちにサービスを受けられるようになる。これは、逆にサービスを提供する側から見れば、新しいサービスを実現する技術を開発した段階で、ネットワーク上でプラグインを配布することで、プラグインを記録媒体に格納して、市の店舗でソフトウェアを販売するという先行投資をすることなく、少ない資本で短期間にサービスを開始できるという利点がある。

【0006】一方、プログラマブルロジックと呼ばれる任意の論理回路を自由に構成することが可能な素子が開発されている。現在、半導体技術の進歩により、比較的規模の大きな論理回路も構成できるようになっている。たとえば、bit Vol.19, No.10, pp.1320-1328にある記こと「シリコンコンバイラー論理合成とその将来」には、プログラマブルロジックを搭載したコンピュータによりプログラムの一部をハードウェア的に実現し、高速に処理するというアイデアが示されている。これは具体的には特開平4-175974号「コプロセッサ論理回路自動生成方法」に示される方法や特開平8-69447号「データ処理装置」に示される方法と同様な方法を用いることにより実現することができる。さらにbit Vol.27, No.10, pp.11-21「FPGAとそのインパクト」に示されるように、プログラマブルロジックを積極的にコンピュータの中心部品として利用し、アプリケーション毎にそのアプリケーションで採用されるアルゴリズムにあった構成のハードウェアに再構成して処理を行うコンピュータシステムも研究されている。

【0007】

【発明が解決しようとする課題】従来の技術に示したネットワークコンピュータを前提とした環境においては、ネットワークの各クライアントである「ネットワークコンピュータ」を利用するユーザは、ネットワーク上で新しいサービスが始まった場合、新しいプラグインをインストールする、あるいは新しいサービスに対応したアプリケーションを利用することにより、「そのサービスを享受することができる。しかし、ここでの大きな問題は、もともと安価にするために高性能なCPUを内蔵せず、ハードウェアの拡張性を持たないネットワークコンピュータでは、計算量を必要とする、より高度なサービスを受けるためには、より高性能なコンピュータに買い換えるか、機能拡張を行うためのハードウェアを追加するかを行う必要がある。また、新しいサービスを開始する側

から見ると、ユーザの持つコンピュータの性能にバラツキがあるため、多くのユーザを獲得するためには平均的なコンピュータの性能で足りる範囲の計算量のサービスしか始められることになり、ネットワーク上の新しいサービスが限られてしまうことになる。

【0008】本発明の目的は、利用者が、より高性能なコンピュータに買い換えるか、機能拡張を行うためのハードウェアを追加するかを行うことなしに、計算量を必要とする、より高度なサービスを受けられるようにすることにある。

【0009】

【課題を解決するための手段】本発明のネットワークコンピュータは以下のように構成される。

【0010】ネットワークにより接続された複数のコンピュータにより構成され、そのうちの少なくとも1つのコンピュータがアプリケーションプログラムを配布するサーバであり、残りのコンピュータが前記アプリケーションプログラムをダウンロードし、実行するクライアントとなるコンピュータであるシステムにおいて、複数の前記クライアントの一部に、プログラムにより機能を随時変更し再構成することが可能な拡張ハードウェアを搭載する。

【0011】そして、サーバに格納されるアプリケーションプログラムが、その一部の機能に対して、前記拡張ハードウェアのプログラムコードとクライアントのメインのプロセッサのコード、の双方のコードを含むようになる。

【0012】さらに、個々のクライアントのOSが各クライアントに前記拡張ハードウェアが実装されているか否かを判断し、前記アプリケーションプログラムの中からハードウェア構成に適したコードのみ取り出す機能を持つ。これにより、アプリケーションプログラムがクライアントにダウンロードされた際に、クライアントのハードウェア構成に適したコードを取り出し実行することで、再構成可能な拡張ハードウェアを持つクライアントのユーザは、より高速にアプリケーションを利用することが可能となるとともに、再構成可能な拡張ハードウェアを持たないユーザもメインプロセッサのコードが選択されることでそのアプリケーションを利用することが可能になる。

【0013】同じ効果は、次のようにしても得られる。
【0014】拡張ハードウェアで実現する機能を、後から動的に追加／削除が可能なクライアント上のOSの拡張機能として実現し、前記のアプリケーションプログラムと同様に拡張ハードウェア用の拡張コードとメインプロセッサコードの2種を含むようにする。そして、アプリケーションプログラム起動時にOSに対し、処理中に利用する前記OSの拡張機能の種類を登録し、前記OSは既にその拡張機能が、前記クライアント上に存在する場合にはそれを用い、存在しない場合はネットワーク上

のサーバから必要とする拡張機能を転送し利用する。

【0015】あるいは、拡張ハードウェアで実現する機能を、実行時にアプリケーションプログラムに動的に追加されるライブラリとして実現し、前記アプリケーションプログラムと同様に拡張ハードウェア用の拡張コードとメインプロセッサコードの2種を含むようにする。そして、アプリケーションプログラムが起動時にOSに対し、処理中に利用する前記動的ライブラリの種類を登録し、前記OSは既にその動的ライブラリが、前記クライアント上に存在する場合にはそれを用い、存在しない場合はネットワーク上のサーバから必要とする動的ライブラリを転送し利用する。

【0016】また、メインプロセッサ用のコード、拡張ハードウェア用のコードが一体となっているのではなく、アプリケーションプログラムまたはOSの拡張機能または動的ライブラリ毎に、個々のコードをホストコンピュータ上に備えても同様の効果が得られる。

【0017】一方、拡張ハードウェアをプログラムを実行するプロセッサにより構成し、拡張コードを前記プロセッサの命令コードとするか、拡張ハードウェアをプログラマブルロジックを用いて構成し、拡張コードを前記プログラマブルロジックを再構成するためのコンフィギュレーション・データとしても同様の効果が得られる。

【0018】拡張ハードウェアがプログラマブルロジックで構成され、その構造がクライアント間で異なる場合に、拡張コードを、適当なゲート数と入出力端子数の論理回路の機能をブール式等で記述した基本モジュールとそれらの接続関係を表現したコードとし、前記基本モジュールをそれぞれプログラマブルロジックの基本プログラム単位に割り付ける機能を持つことで複数の構造に対応したコードを実現できる。また、複数のプログラマブルロジックチップにまたがる大きな拡張コードの場合は、基本モジュールを接続の度合に応じて分割し、各プログラマブルロジックチップに配置配線し端子の機能を決定する機能と、プログラマブルロジック内の基本プログラム単位への基本モジュールの配置配線をする機能をサーバまたはクライアント上に持つことで対応する。

【0019】これと同じ効果は、拡張コードを複数の構造の個々のコードをすべて含むようにし、ハードウェア構造に合わせたコードを選択する機能をコード選択機能が持つことにより実現できる。

【0020】また、以上の効果は、メインプロセッサをプログラマブルロジックで構成し、プログラマブルロジックの利用されていない論理資源を拡張ハードウェアとして利用することでも得られる。

【0021】クライアントのOSは、拡張ハードウェアを利用する複数のアプリケーションを同時に実行するために、各アプリケーションが利用する拡張ハードウェアのハードウェア資源を管理し、必要ななくなったハードウェア資源を別のアプリケーションプログラムのために

再利用し、各アプリケーションをメインプロセッサで実行するか、拡張ハードウェアで実行するかを決定する拡張ハードウェア管理機能と前記拡張ハードウェアに入り切らないアプリケーションプログラムの拡張コードを時分割で入れ替えるコード入れ替え機能を持つ。

【0022】また、拡張ハードウェア管理機能は、クライアント上で実行されるアプリケーションプログラム毎に適宜設定される実行プライオリティ値、前記アプリケーションプログラムをすべてメインプロセッサで処理するために必要とする正規化されたメインプロセッサの処理能力値、前記アプリケーションプログラムを拡張ハードウェアで処理する場合に必要な正規化されたメインプロセッサと拡張ハードウェアの処理能力値、前記アプリケーションプログラム毎に必要とする拡張ハードウェアの正規化されたハードウェア資源使用量、前記クライアントのメインプロセッサの正規化された処理能力値、前記クライアントの拡張ハードウェアの正規化された処理能力値、前記クライアントの拡張ハードウェアの正規化されたハードウェア資源量、および前記クライアントの拡張ハードウェアのコードを入れ替えるために必要な正規化された処理能力値を元に、前記クライアントの拡張ハードウェアの持つハードウェア資源に入り切らない複数のアプリケーションプログラムに対して、優先的に拡張ハードウェアを利用するアプリケーションプログラム、時分割で拡張コードを入れ替えて使用するアプリケーションプログラム、メインプロセッサ用コードで処理するアプリケーションプログラムを選択する。

【0023】拡張ハードウェアが、複数の内部状態を同時に保持し、かつそれらを切り替える機能を持ち、内部状態のみを時分割で切り替えて機能を共有することにより、複数のアプリケーションが同時に同じ拡張コードを拡張ハードウェアで利用できる。

【0024】また、拡張ハードウェアが、複数の内部状態と複数の構成データを同時に保持し、かつそれらを切り替える機能を持ち、内部状態と構成データを時分割で切り替えて使用することにより、複数のアプリケーションが同時に複数の拡張コードを拡張ハードウェアで利用できる。

【0025】さらに、拡張ハードウェアが、複数の内部状態と複数の構成データを同時に保持し、かつそれらが外部から与えられる識別番号と、あらかじめ構成データ、内部状態毎に割り当てられた識別番号と一致する部分のみが選択される機能を持つことにより、拡張ハードウェア内に実際に存在する内部状態、構成データの数よりも多くのハードウェア機能を実現できる。

【0026】

【発明の実施の形態】

<実施例1>図1は本発明のコンピュータシステムの一例である。本コンピュータシステムはサーバと、複数個のクライアントによって構成される。図1ではサーバは

102クライアントは103と104である。サーバとクライアントはそれぞれネットワーク112, 113で接続されており、相互にメッセージやデータの送受信をすることができる。

【0027】サーバ102はクライアントに向けて提供するためのクライアント上で動作可能なアプリケーションプログラム108を保有している。

【0028】クライアント103は、メインプロセッサ105を備えており、さらに拡張ハードウェア106を搭載している。一方、クライアント104は、メインプロセッサ107を備えているが、拡張ハードウェアは搭載していない。

【0029】クライアント103, 104はサーバ102からアプリケーションプログラムを自機上にそれぞれ読み込み(116, 118)、実行することができる。またそれぞれのハードウェアに適したコードをアプリケーションプログラム116, 118から選択する、コード選択機能123, 124をそれぞれのOS121, 122に備えている。

【0030】アプリケーションプログラム108はその一部として拡張ハードウェア106上で動作可能なコードである拡張コード109と、メインプロセッサ105または107上で動作可能なメインプロセッサコード110を含んでいるため、ハードウェア構成が異なるクライアント上でそのアプリケーションを実行することができる。

【0031】図1においてクライアント103ではコード選択機能123が拡張コード115を選択し、クライアント104ではコード選択機能124がメインプロセッサコード120を選択し実行する様子を示している。

【0032】また、図1のコード選択機能123, 124と同様な機能をサーバ102上に備え、アプリケーションプログラムを転送する先のクライアントのハードウェア構成に合わせたコードを選択し、転送することでも図1と同等のコンピュータシステムを実現することができる。

【0033】次に図2から図6を用いて、図1で示したコンピュータシステム上で、あるアプリケーションプログラムを提供する様子を説明する。

【0034】図2ではサーバ102はアプリケーションプログラムとして3次元CG処理を行うプログラム208を保持している。3次元CGプログラム208は拡張コード209とメインプロセッサコード210を含んでいる。矢印230はクライアント103がサーバ102に対して必要なアプリケーションプログラムの要求を発行する様子を示し、矢印231はサーバ102がクライアント103に3次元CG208を転送する様子を示している。3次元CGプログラム214は3次元CGプログラム208がクライアント103上に読み込まれたものである。

【0035】図3は図2の3次元CGプログラム208の拡張コード209の1つの機能を示している。図3の式300は3次元の座標値(x, y, z)を行列301によって変換する3次元座標変換関数である。この機能を拡張ハードウェア106上で行うことで3次元の物体の移動、回転などの計算を高速に行うことができる。

【0036】次に図4、図5、図6を用いて図2のコンピュータシステムの動作を説明する。

【0037】図4はクライアント103の動作を示したPADである。最初のステップ401ではクライアント103のOS121は、サーバ102に対して必要なアプリケーションプログラム3次元CG208の要求を発行する。ステップ401は図2の矢印230で示した動作に対応する。

【0038】次にステップ402でクライアント103はサーバ102からのデータ受信待ち状態に入り、サーバ102が送信する3次元CGプログラム208をロードする。ロードされたプログラムが214である。ステップ402は図2の矢印231で示した動作に対応する。

【0039】ステップ403のコード選択機能では、OS121が、ロードした3次元CGプログラム214からクライアント103のハードウェア構成に適したコードを選択して拡張ハードウェア106に準備する。図2では拡張コード215である3次元座標変換関数300を拡張ハードウェア106上に準備する。ステップ403の詳細は後述する。

【0040】ステップ404では3次元CGプログラム214を実行する。このとき3次元CG処理の大部分はメインプロセッサ105上で実行するが、拡張コード215で表現されている3次元座標変換関数300は拡張ハードウェア上で実行される。このため3次元座標変換が高速に実行できる。

【0041】図5はサーバ102の動作を示したPADである。最初のステップ501はクライアント103からのアプリケーションプログラムの要求を待つループである。

【0042】ステップ502はクライアント103が3次元CGプログラム208の転送要求を発行した場合にステップ503に進む、また要求が発行されない場合はステップ501でループする。

【0043】ステップ503はクライアント103の要求に従って3次元CGプログラム208を送信する。

【0044】図6はクライアント103の機能の一部であるコード選択機能600(図4のステップ403)の動作を示した図である。最初のステップ601はクライアント103のハードウェア構成に従った分岐である。103が拡張ハードウェア106を持つ場合には602に進み、持たない場合には605に進む。

【0045】ステップ602は拡張ハードウェア106の種類に従った分岐である。例えば拡張ハードウェアの

11

種類がプログラマブルロジックである場合にはステップ603に進み、汎用プロセッサである場合には604に進む。その他の拡張ハードウェアとして搭載可能なプロセッサについても本ステップの分岐を拡張することで対応できる。

【0046】ステップ603ではアプリケーションプログラムからプログラマブルロジック用のコードを選択する。図2の例ではこのステップで3次元CGプログラム214の一部として拡張コード215を選択する。

【0047】ステップ604ではアプリケーションプログラムから汎用プロセッサ用のコードを選択する。 10

【0048】ステップ605ではアプリケーションプログラムからメインプロセッサ用のコードを選択する。

【0049】以上のサーバ102とクライアント103の動作の相互作用により、サーバ102上の3次元CGプログラム208はクライアント103上にロードされる。さらにアプリケーションプログラムはメインプロセッサ105と拡張ハードウェア106上で実行することができる。

【0050】<実施例2>図7は本発明の第2の実施例である。図1とはサーバ102が保有するアプリケーションプログラムの構成が異なる。図1のアプリケーションプログラム108は拡張コード109とメインプロセッサコード110を含んでいたが、図7では拡張ハードウェア上で実現する機能は拡張OS725として保有する。拡張OS725は拡張コード709、メインプロセッサコード710を含んでおり、図1のアプリケーションプログラム108と同様に、クライアント103がロード可能なOSとして提供される。

【0051】クライアント103ではアプリケーションプログラム714の起動に際し、必要な拡張OSが不足している場合にはサーバ102から必要な拡張OSをロードし、拡張ハードウェア106上で実行する。

【0052】<実施例3>図8は本発明の第3の実施例である。図1とはサーバ102が保有するアプリケーションプログラムの構成が異なる。図8では拡張ハードウェア上で実現する機能は動的ライブラリ825として保有する。動的ライブラリ825は拡張コード809、メインプロセッサコード810を含んでおり、図1のアプリケーションプログラム108と同様に、クライアント103がロード可能な動的ライブラリとして提供される。

【0053】クライアント103ではアプリケーションプログラム814の起動に際し、必要な動的ライブラリが不足している場合にはサーバ102から必要な動的ライブラリをロードし、拡張ハードウェア106上で実行する。

【0054】<実施例4>図9は本発明の第4の実施例である。図1とはサーバ102が保有する、アプリケーションプログラムの構成が異なる。図9ではメインプロ

12

セッサのみで動作するアプリケーションプログラム909と拡張ハードウェアに対応したアプリケーションプログラム910をそれぞれ保持し、それをクライアント103がロード可能なプログラムとして提供する。

【0055】クライアント103はそのハードウェア構成に合ったアプリケーションプログラムをロードし、拡張ハードウェア106上で実行する。またアプリケーションプログラムは図7、図8と同様に拡張OS、または動的ライブラリであっても同様の機能が実現できる。

【0056】<実施例5>また、上記した図1、図7、図8、図9の実施例は拡張ハードウェア106をプログラムを実行する汎用プロセッサまたはプログラマブルロジックにより構成することもできる。

【0057】<実施例6>図10は第6の実施例の構成を示した図である。図10では拡張ハードウェア106はプログラマブルロジックである。図1とは拡張コード1009が異なる。図10の拡張コードはプログラマブルロジック用のコードであるが、異なった構成のプログラマブルロジックに対応できるように抽象度の高い記述になっている。拡張コード1009をクライアント103上にロードしたものが、1015である。図10では拡張コード1015を拡張ハードウェア106のプログラマブルロジックの構成に合うように変換するコード変換機能1023を備えている。

【0058】図11は図10の拡張コード1015の例である。1104、1105は適当な大きさのゲート数と入力端子数の論理回路の機能をブール式（論理演算式）や算術演算式などの抽象的な記述で表わした基本モジュールの例である。基本モジュールをいくつか定義し、さらに基本モジュール間の接続関係を1106に定義することで、拡張コードを記述する。1101は基本モジュールの個数を定義する記述、1102は拡張コード1015の外部からの入出力数を定義する記述、1103は外部からの入出力と基本モジュールとの接続を定義する記述である。

【0059】図12は図11の拡張コード1015が基本モジュールとその接続関係で表現される様子を図で表現したものである。

【0060】図13は拡張ハードウェア106のプログラマブルロジックの基本プログラム単位を示したものである。基本プログラム単位1300は一定数の入出力端子と論理素子を持つ論理回路のブロックである。図13は4入力1出力で再構成可能論理1303はコンフィグレーションメモリ1301の内容により、任意の論理関数を実現できる。また内部状態レジスタ1304は状態を保持するフリップフロップである。プログラマブルロジックでは基本ブロックがアレー状に並んでいる。基本プログラム単位内、基本プログラム単位間の配線は、書き換え可能なメモリに格納された配線データによって決定される。配線データは基本ブロック間の配線の有無を

0または1で示したビットパターンで表現される。

【0061】図14を用いてコード変換機能1023の処理を説明する。コード変換機能1023の入力は拡張コード1015である。出力はプログラマブルロジックの配線データである。

【0062】ステップ1402は基本モジュール1104をプログラマブルロジックの基本プログラム単位1300に割り当て、1300内部のコンフィグレーションメモリに設定するデータを決定する。他のすべての基本モジュールについても同様の処理を行う。

【0063】ステップ1403は基本ブロック単位に割り付けの済んだ基本モジュールを複数のプログラマブルロジック・チップのどれに割り付けるかを決定する。

【0064】ステップ1404はプログラマブルロジック・チップ間の配置配線を決定し、端子の割り当てを決める。

【0065】ステップ1405はプログラマブルロジック・チップ内の配置配線を決定する。

【0066】以上の各ステップで決定された配置配線は配線データとして書き換え可能なメモリに格納される。

【0067】<実施例7>図6において602の選択機能に、プログラマブルロジックやプロセッサの種類毎の選択を可能とすることにより、拡張ハードウェアの構造の種類に応じてコードを選択し実行することができる。

【0068】<実施例8>図15は第8の実施例の構成を示した図である。図15では、図1、図7、図8、図9、図10におけるメインプロセッサ105の機能が、拡張ハードウェアの一部のハードウェア資源を用いて構成されており、残りのハードウェア資源を拡張ハードウェア106として利用する。このような構成においても、上記各図の実施例に示したシステムを構成することができる。

【0069】<実施例9>図16は第9の実施例の構成を示した図である。図16において、103は図1におけるクライアント103と同じ構成であり、メインプロセッサ105と拡張ハードウェア106を持つ。121はOSであるが、ここで、図1とは異なり、コード選択機能123の他に、拡張ハードウェア管理機能1601、拡張コード入れ替え機能1603を持つ。またクライアント103では複数のアプリケーションプログラムを実行できる。ここではアプリケーションプログラムA、1604とアプリケーションプログラムB、1607を実行する例を示す。

【0070】アプリケーションが終了した時、アプリケーションが新たに始まる時、ユーザの指示があった時のいずれかにアプリケーション毎のプライオリティーPiが決定され、拡張ハードウェア管理機能1601が起動し、各アプリケーションをメインプロセッサ105で実行するか、拡張ハードウェア106で実行するか、あるいは拡張ハードウェアで実行するが、時分割で内容を入れ替えるかを決定する。

れ替えながら実行するかを決定する。

【0071】図17は上記拡張ハードウェア管理機能1601の動作を説明する図である。図中で、iはアプリケーションプログラムi、cはクライアントc、MPはメインプロセッサ、EHWは拡張ハードウェア、Piはiに設定される実行プライオリティ値、MmiはiをMPだけで処理するために必要とする正規化されたMPの処理能力値、Ms iはiをEHWで処理する場合に必要な正規化されたMPの処理能力値、Me iはiをEHWで処理する場合に必要な正規化されたEHWの処理能力値、Re iはiが必要とするEHWの正規化されたハードウェア資源使用量、MmcはcのMPの正規化された処理能力値、Mc cはcのEHWの正規化された処理能力値、Rec cはcのEHWの正規化されたハードウェア資源量、Dmc cはcのEHWのコードを入れ替えるために必要な正規化された処理能力値、Smm cは現在cのMP上で実行している処理の必要処理能力量の総和、Srec cは現在cのEHW上で処理しているハードウェア資源量の総和、Nd cは現在cでコードを入れ替えるにより処理している拡張コードの総数である。

【0072】図17では、まず総和値Smm c、Srec c、Nd cを初期化する(1701)。次に、現在実行中の各アプリケーションiについて、Piの高い順に1703～1710を繰り返す。1703の条件分岐は、メインプロセッサで処理できる限り、または拡張ハードウェアで実行するより、メインプロセッサで実行した方が高速に処理できる場合に1704～1705へ、そうでない場合は1706～1710へ分岐する。1704～1705はアプリケーションiをメインプロセッサで実行する様に設定する。1706の条件分岐は拡張ハードウェアのハードウェア資源が満杯になっているかどうかを判定し、まだ余裕がある場合には1707～1708へ、余裕がない場合は1709～1710へ分岐する。1707～1708ではアプリケーションiを拡張ハードウェアで実行する様に設定する。1709～1710ではアプリケーションiの拡張コードを時分割で入れ替えて実行する様に設定する。このとき、既に拡張ハードウェアで実行することを設定したアプリケーションの中でiに入れ替えてもハードウェア資源が満杯にならず、最もプライオリティが低いアプリケーションjを入れ替え対象とする。もし、そのような単独のjが存在しなければ、2つ以上のアプリケーションのペアをjとする。以上の処理により、現在実行中の複数のアプリケーションをどのような形態で実行するかを決定する(図16において1610、1611)。

【0073】次に、コード選択機能123は1601で決定された実行方法に従って各アプリケーションのコードを選択する(1612、1613)。また、アプリケーションプログラム実行時に拡張コード入れ替え機能1603が時分割の処理単位時間毎に入れ替え対象のア

リケーションの拡張コードを入れ替える(1615, 1616)。

【0074】<実施例10>図18は、複数の内部状態を同時に保持し、かつそれらを切り替える機能を持つプログラマブルロジックの構成を説明する図である。このプログラマブルロジックは内部状態のみを時分割で切り替えて使用することにより、複数のアプリケーションが同時に同じ拡張コードを拡張ハードウェアで利用する場合に適している。図18において1801はプログラマブルロジックの基本プログラム単位である。1802は再構成可能な論理回路部分で、コンフィグレーション・メモリ1803によりその機能が定義される。1804と1805は回路の内部状態を保持するためのレジスタである。このレジスタの値への書き込み信号と出力信号は内部状態切り替え線1807により切り替わる。

【0075】<実施例11>図19は、複数の内部状態と複数の構成データを同時に保持し、かつそれらを切り替える機能を持つプログラマブルロジックの構成を説明する図である。このプログラマブルロジックは内部状態と構成データ(コンフィグレーションメモリの内容)を時分割で切り替えて使用することにより、拡張ハードウェア上に、同時に複数の機能を時分割で実現する場合に適している。図19において1901はプログラマブルロジックの基本プログラム単位である。1902は再構成可能な論理回路部分で、コンフィグレーション・メモリ1903, 1904によりその機能が定義される。また1906と1907は回路の内部状態を保持するためのレジスタである。コンフィグレーション・メモリと内部状態レジスタは論理切り替え線1909により切り替わる。

【0076】<実施例12>図20は、複数の内部状態と複数の構成データを同時に保持し、かつそれらが外部から与えられる識別番号と、あらかじめ構成データ、内部状態毎に割り当てられた識別番号と一致する部分のみが選択される機能を持つプログラマブルロジックの構成を説明する図である。このプログラマブルロジックは切り替えて使用できるコンフィグレーションメモリと内部レジスタの数よりも多くのハードウェア機能を時分割で実現する場合に適している。図20において2001はプログラマブルロジックの基本プログラム単位である。2002は再構成可能な論理回路部分で、コンフィグレーション・メモリ2003, 2004によりその機能が定義される。また2007と2008は回路の内部状態を保持するためのレジスタである。論理切り替え線2011により与えられる論理切り替え識別番号は、各コンフィグレーション・メモリと内部状態レジスタ毎に存在する識別番号を記憶した比較回路2005, 2006, 2007, 2008により比較され、内容が一致したコンフィグレーションメモリと内部状態レジスタが使用される。

【0077】

【発明の効果】本発明により、ネットワークで接続されたコンピュータ上で、サーバから配布されたアプリケーションプログラムをクライアント側で実行する際に、アプリケーションプログラムに合わせて、クライアント側の計算機の構成を変え、処理に適した構成としてことで、アプリケーションプログラムを高速に処理できるようになる。また、従来、ネットワーク上で、クライアント側に特殊なハードウェアを必要とする新しいサービスを開始しようとする場合、クライアント側のユーザはそのためには新しいクライアントを導入するか、特殊なハードウェアをクライアント上に導入する必要があったし、サービスの提供者は、新しいクライアントあるいは特殊なハードウェアを持つごく一部のユーザに対してのみ新しいサービスを提供することになった。しかし、本発明を実施することにより、特殊なハードウェアを導入することなく、新しいサービスを開始することが可能となる。

【0078】また、本発明によればクライアント側の拡張ハードウェアの有無に関係なく、あるいは拡張ハードウェアの構造に関係なくユーザは所望のアプリケーションを実行できる。

【0079】さらに、本発明により1つのクライアント上で、同時に複数のアプリケーションプログラムを実行する場合に拡張ハードウェアのハードウェア資源を有効に利用でき、またそのような機能を実現するのに適した拡張ハードウェアに使用するプログラマブルロジックの構成を提供する。

【図面の簡単な説明】

- 30 【図1】本発明の一実施例の構成の説明図。
- 20 【図2】本発明の一実施例の動作の説明図。
- 30 【図3】拡張ハードウェアで処理する機能の説明図。
- 40 【図4】クライアントの動作の説明図。
- 50 【図5】サーバコンピュータの動作の説明図。
- 40 【図6】クライアントの構成に合わせてコードを選択するコード選択機能の説明図。
- 50 【図7】拡張OSにより拡張ハードウェアを利用する場合のシステム構成の説明図。
- 40 【図8】動的ライブラリにより拡張ハードウェアを利用する場合のシステム構成の説明図。
- 50 【図9】ハードウェア構成に合わせたアプリケーションプログラムの中からクライアントの構成に合わせたコードを転送する場合のシステム構成の説明図。
- 40 【図10】拡張コードを拡張ハードウェアの構造に合わせて変換するコード変換機能を持つ場合のシステム構成の説明図。
- 50 【図11】コード変換機能を持つ場合の拡張コードの説明図。
- 50 【図12】図11の拡張コードの構造の説明図。
- 50 【図13】拡張ハードウェアを構成するプログラマブル

17

ロジック内の基本プログラム単位の説明図。

【図14】図10のコード変換機能の動作の説明図。

【図15】拡張ハードウェア中にメインプロセッサ機能を実現したクライアントの構成の説明図。

【図16】複数のアプリケーションを同時に実行する場合の構成の説明図。

【図17】複数のアプリケーションを同時に実行する場合に用いる拡張ハードウェア管理機能の動作の説明図。

【図18】内部状態を切り替える機能を持つプログラマブルロジックの構成の説明図。

*10

18

* 【図19】コンフィグレーション・メモリと内部状態を切り替える機能を持つプログラマブルロジックの構成の説明図。

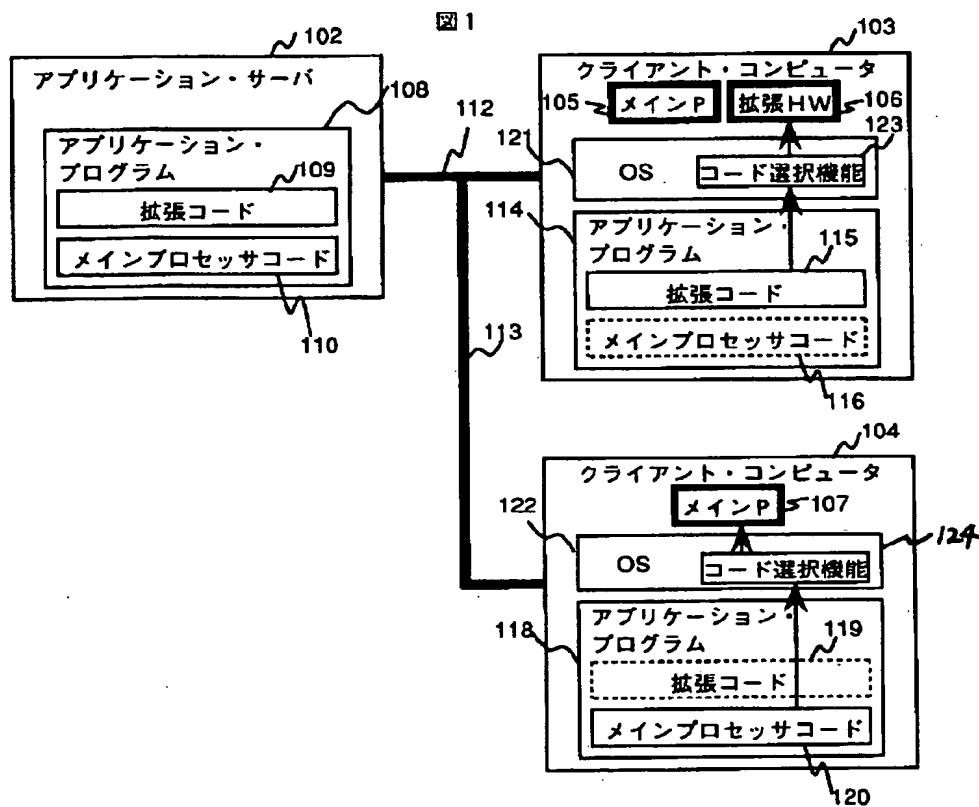
【図20】コンフィグレーション・メモリと内部状態を外部から与える識別番号により切り替える機能を持つプログラマブルロジックの構成の説明図。

【符号の説明】

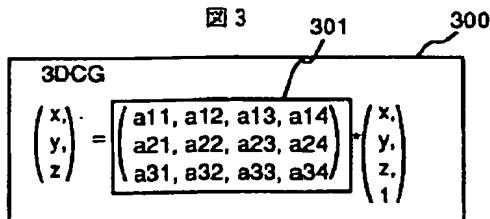
105…メインプロセッサ、106…拡張ハードウェア、112, 113…通信ネットワーク。

*10

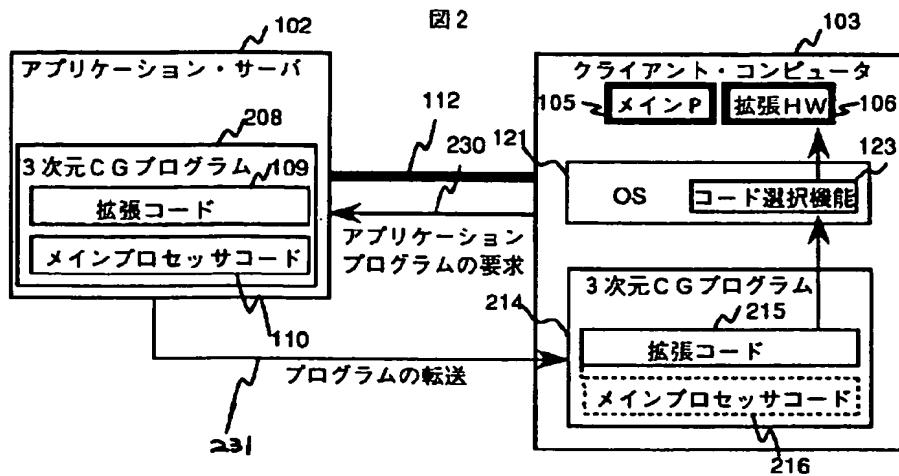
【図1】



【図3】



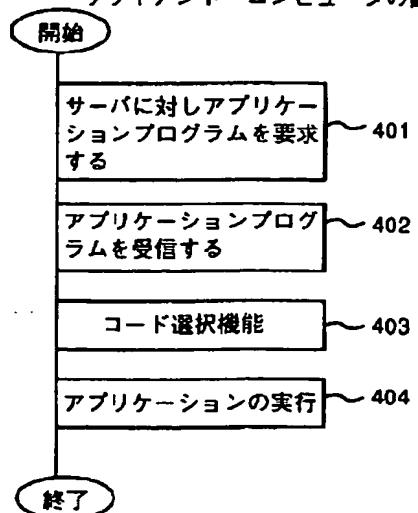
【図2】



【図4】

図4

クライアント・コンピュータの動作 400



【図11】

図11

拡張コード 1015

```

Number of Basic Modules: 9 ~ 1101
Number of Primary Inout: 8 ~ 1102
{ PIO01 = BM01.IO01, PIO02 = ~ 1103
BM01.IO02, ... }
Definitions of Basic Module s: {

```

```

BM01: {
O01 = .I(BM01.IO01,...);
O02 = ....;
}

```

基本モジュール 1104


```

BM02: {
O01 = .I(BM02.IO01,...);
O02 = ....;
}

```

基本モジュール 1105

```

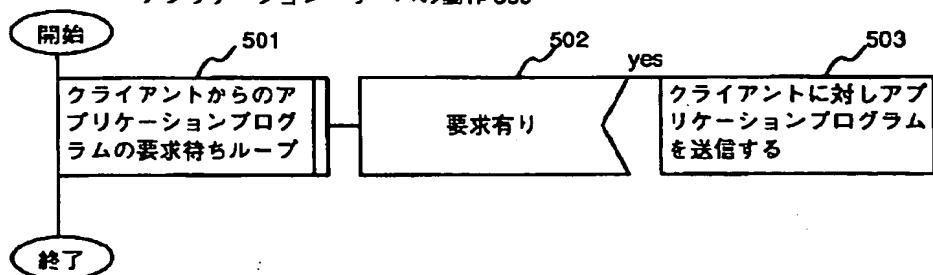
}
Definitions of Network: { ~ 1106
}

```

【図5】

図5

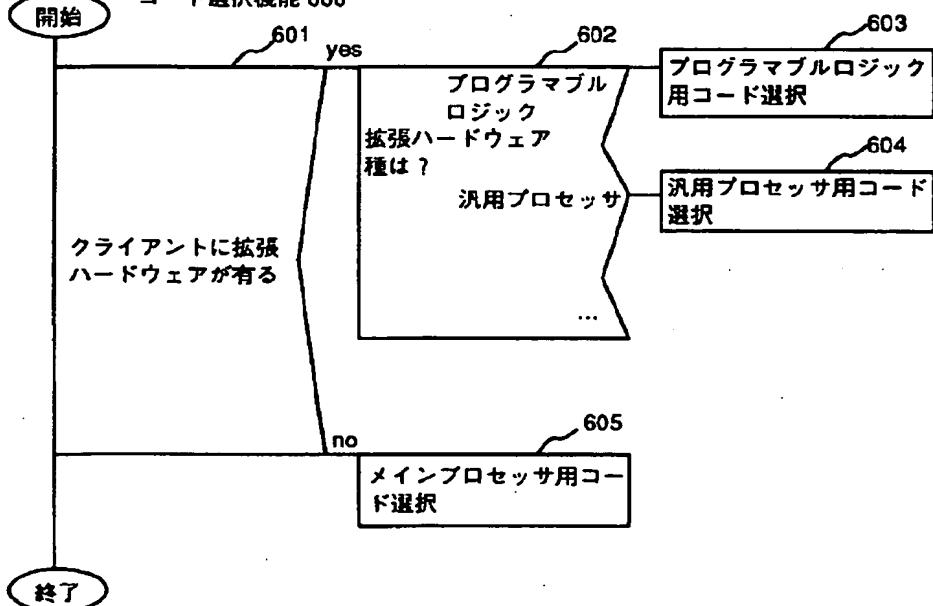
アプリケーション・サーバの動作 500



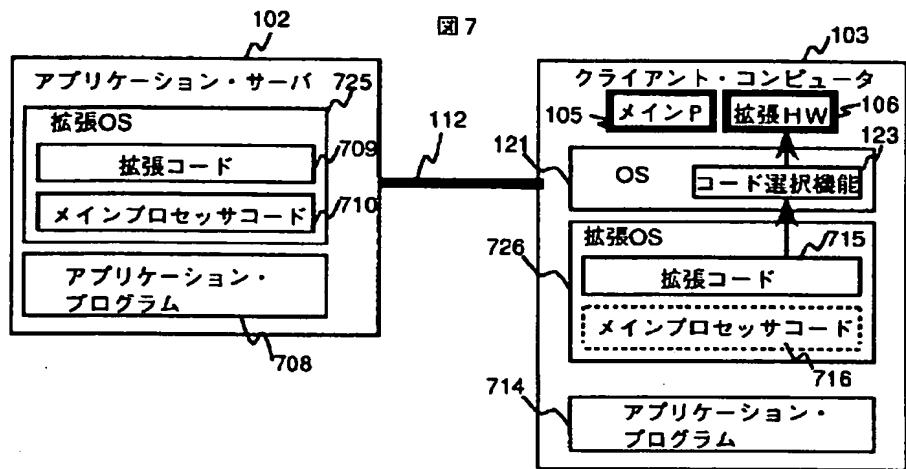
【図6】

図6

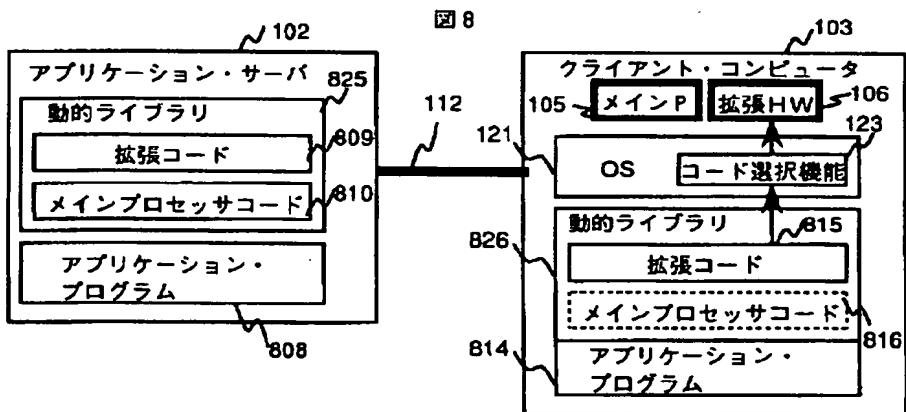
コード選択機能 600



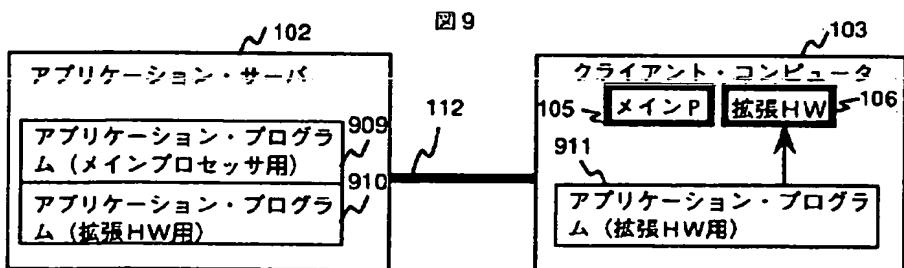
【図7】



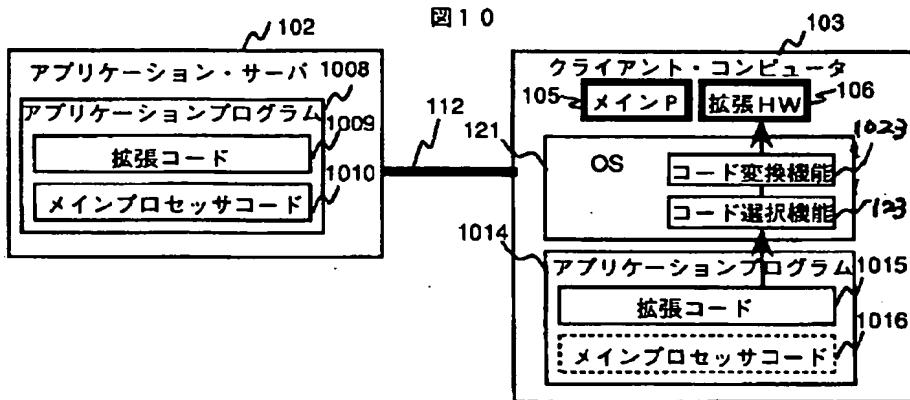
【図8】



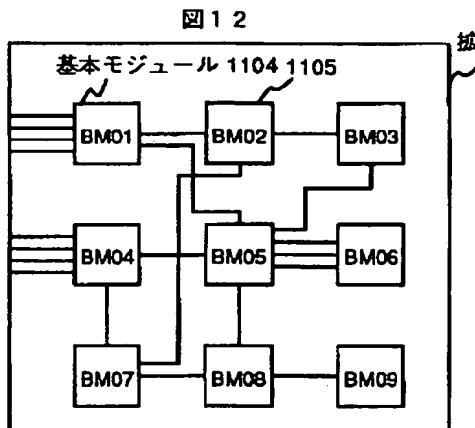
【図9】



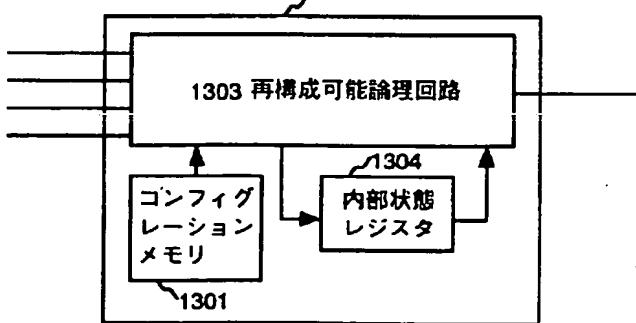
【図10】



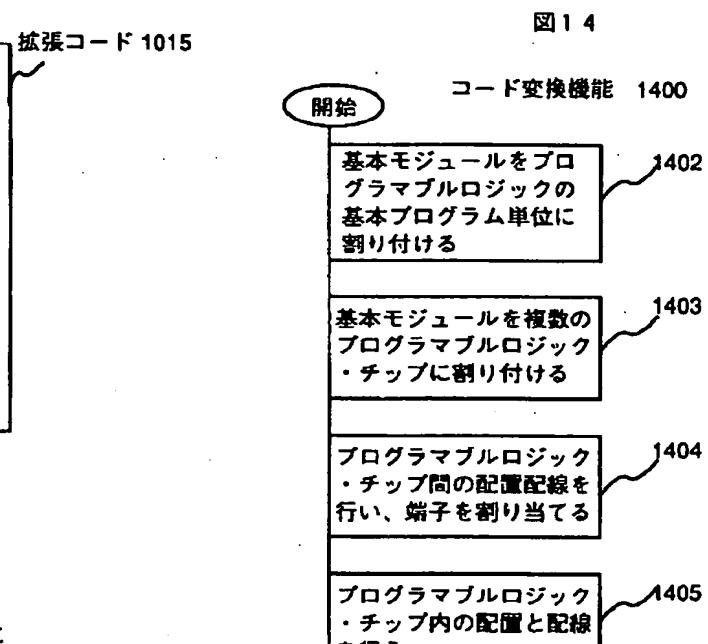
【図12】



【図13】

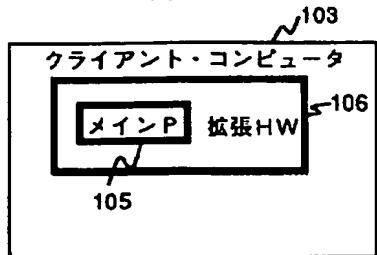
図13
1300 基本プログラム単位

【図14】



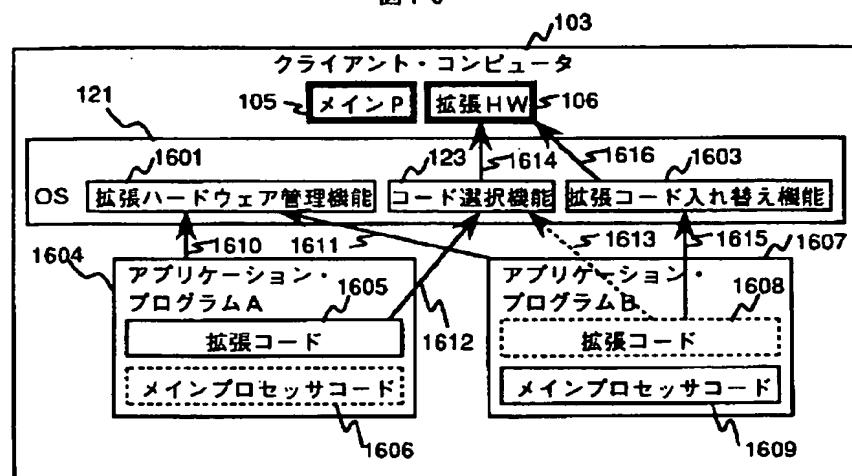
【図15】

図15



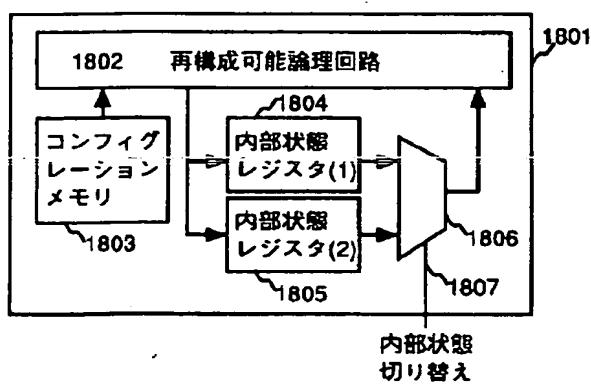
【図16】

図16



【図18】

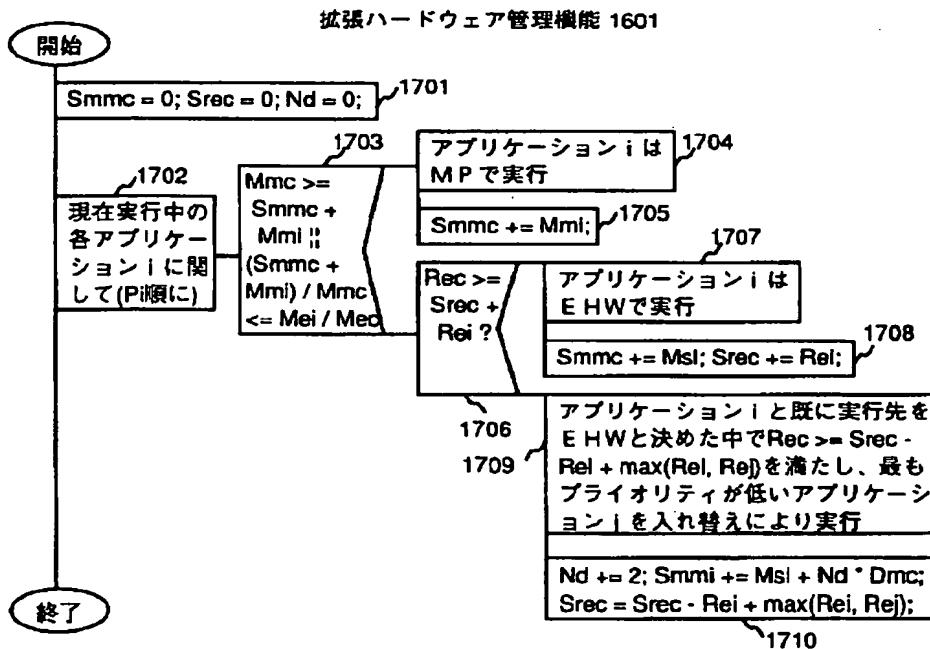
図18



【図17】

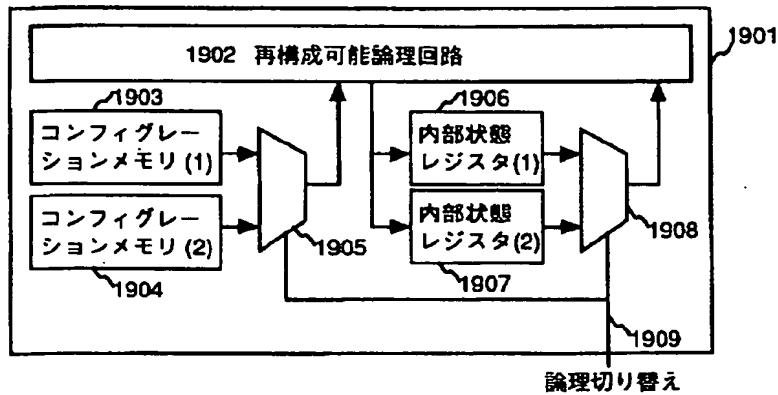
図17

i : アプリケーション プログラム i
 c : クライアント・コンピュータ c
 MP : メインプロセッサ
 EHW : 拡張ハードウェア
 Pi : i に設定される実行プライオリティ値
 Mmi : i を MP だけで処理するために必要とする正規化された MP の処理能力値
 Msi : i を EHW で処理する場合に必要な正規化された MP の処理能力値
 Mei : i を EHW で処理する場合に必要な正規化された EHW の処理能力値
 Rei : i が必要とする EHW の正規化されたハードウェア資源使用量
 Mmc : c の MP の正規化された処理能力値
 Mec : c の EHW の正規化された処理能力値
 Rec : c の EHW の正規化されたハードウェア資源量
 Dmc : c の EHW のコードを入れ替えるために必要な正規化された処理能力値
 Smmc : 現在 c の MP 上で実行している処理の必要処理能力量の総和
 Srec : 現在 c の EHW 上で処理しているハードウェア資源量の総和
 Nd : 現在 c でコードを入れ替えにより処理している拡張コードの数



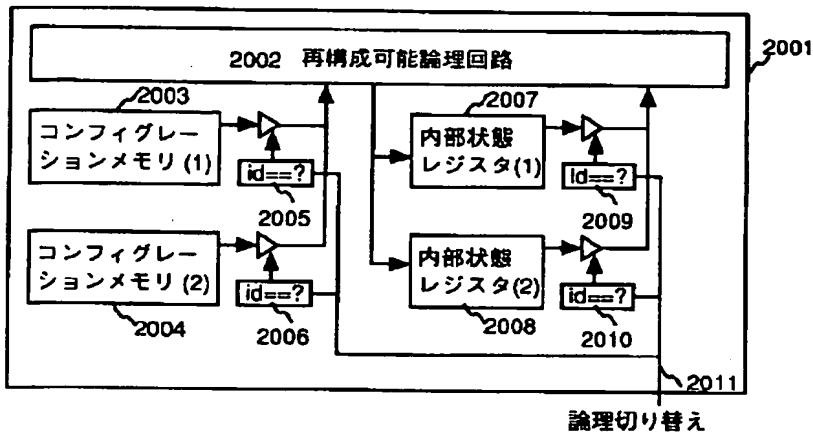
【図19】

図19



【図20】

図20



【公報種別】特許法第17条の2の規定による補正の掲載

【部門区分】第6部門第3区分

【発行日】平成15年8月15日(2003.8.15)

【公開番号】特開平10-78932

【公開日】平成10年3月24日(1998.3.24)

【年通号数】公開特許公報10-790

【出願番号】特願平8-233913

【国際特許分類第7版】

G06F 13/00 357

9/06 410

9/445

13/14 330

15/00 310

【F1】

G06F 13/00 357 Z

9/06 410 S

13/14 330 A

15/00 310 J

9/06 420 J

【手続補正書】

【提出日】平成15年5月6日(2003.5.6)

【手続補正1】

【補正対象書類名】明細書

【補正対象項目名】特許請求の範囲

【補正方法】変更

【補正内容】

【特許請求の範囲】

【請求項1】ネットワークに接続され、サーバと上記ネットワークを介してアプリケーションプログラムの送信を受け実行するクライアントであって、

機能を変更し、再構成することが可能なハードウェアを有し、

上記サーバから上記アプリケーションプログラムを読み込む際に上記ハードウェア上で動作する第1プログラムコードを読み込み、上記第1プログラムコードに基づき上記ハードウェアの機能を構成して上記アプリケーションプログラムを実行するクライアント。

【請求項2】請求項1において、

メインプロセッサを有し、

上記サーバから上記アプリケーションプログラムを読み込む際に、上記メインプロセッサ上で動作する第2プログラムコードを読み込み、

上記ハードウェアの有無により、上記第1プログラムコ

ード及び上記第2プログラムコードのいずれかを実行するクライアント。

【請求項3】請求項1において、

上記ハードウェアはプログラマブルロジックであって、上記第1プログラムコードは、それぞれ論理回路の機能を記述した複数の基本モジュールと上記基本モジュール間の接続関係の定義を含むクライアント。

【請求項4】サーバからネットワークを介してアプリケーションプログラムを配布するアプリケーションプログラムの配布方法であって、

上記ネットワークに接続されたクライアントからのアプリケーションプログラムの要求を受け、

上記要求に応答して、上記クライアントに上記アプリケーションプログラムを配布し、

上記アプリケーションプログラムを上記配布する際に、機能を変更し、再構成することが可能なハードウェア上で動作する第1プログラムコードを配布するアプリケーションプログラムの配布方法。

【請求項5】請求項4において、

上記アプリケーションプログラムを上記配布する際に、上記クライアントのメインプロセッサ上で動作する第2プログラムコードを配布するアプリケーションプログラムの配布方法。